

***Amendments to the Claims***

This listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Previously Presented) A method in a computer system for flexibly altering software component behavior, the method comprising:
  - intercepting a service request made by a software component;
  - evaluating the intercepted service request based on at least one dynamically alterable condition dependent rule, an original or modified data in the service request, and at least one of a present software system state and a past software system state, wherein the at least one dynamically alterable condition dependent rule is alterable while the requesting software component is running;
  - dynamically selecting at least one desired behavior from among several behaviors for the software component based on the evaluation; and
  - dynamically controlling the software component such that the software component executes the selected desired behavior.
2. (Original) The method of claim 1 wherein:
  - intercepting a service request comprises intercepting a software supported system call.
3. (Original) The method of claim 2 wherein:
  - intercepting a software supported system call further comprises redirecting an entry in an interrupt vector table to alternative code.
4. (Original) The method of claim 1 wherein:
  - intercepting a service request comprises intercepting a hardware supported system call.
5. (Original) The method of claim 4 wherein:

Reply to Office Action of April 14, 2009

intercepting a hardware supported system call further comprises redirecting an entry in an interrupt vector table to alternative code.

6. (Original) The method of claim 1 wherein:

intercepting a service request comprises intercepting a software library based subroutine call.

7. (Original) The method of claim 6 wherein:

intercepting a software library based subroutine call further comprises modifying at least one dynamically linked library.

8. (Original) The method of claim 1 wherein:

intercepting a service request comprises intercepting a subroutine based service.

9. (Previously Presented) The method of claim 8 wherein:

intercepting a subroutine based service further comprises redirecting a subroutine call instruction to alternative code.

10. (Original) The method of claim 8 wherein:

intercepting a subroutine based service further comprises patching machine language entry code of the subroutine.

11. (Original) The method of claim 1 wherein:

intercepting a service request comprises intercepting a service dispatch mechanism based on dynamic name resolution.

12. (Original) The method of claim 11 wherein:

intercepting a service dispatch mechanism based on dynamic name resolution further comprises modifying service lookup name space.

13. (Previously Presented) The method of claim 1 further wherein the selected desired behavior for the software component further comprises:  
executing alternative code in response to intercepting the service request.
14. (Original) The method of claim 13 further comprising:  
executing alternative code in addition to calling the service request.
15. (Original) The method of claim 13 wherein:  
the alternative code performs an operation with a same purpose as that of the service request.
16. (Original) The method of claim 13 wherein:  
the alternative code performs an operation with a different purpose from that of the service request.
17. (Previously Presented) The method of claim 1 further wherein the selected desired behavior for the software component further comprises:  
preventing execution of the service request.
18. (Original) The method of claim 17 further comprising:  
returning a value to the software component so as to simulate execution of the service request, without actually calling the service request.
19. (Previously Presented) The method of claim 1 further wherein the selected desired behavior for the software component further comprises:  
preventing code that executes in response to interception of the service request from accessing at least some data.
20. (Original) The method of claim 19 further comprising:  
allowing code that executes in response to interception of the service request to access alternative data, different from requested data.

21. (Previously Presented) The method of claim 20 wherein:  
the alternative data comprises a copy of at least some requested data.
22. (Original) The method of claim 19 wherein:  
code that executes in response to the interception of the service request comprises  
at least alternative code.
23. (Original) The method of claim 19 wherein:  
code that executes in response to the interception of the service request comprises  
at least the service request.
24. (Previously Presented) The method of claim 1 wherein dynamically controlling  
the software component further comprises:  
passing alternative parameters to code that executes in response to interception of  
the service request.
25. (Original) The method of claim 24 further comprising:  
creating the alternative parameters by modifying original parameters passed to  
the service request.
26. (Previously Presented) The method of claim 1 wherein dynamically controlling  
the software component further comprises:  
executing alternative code in response to interception of the service request.
27. (Previously Presented) The method of claim 1 wherein dynamically controlling  
the software component further comprises:  
executing alternative code in response to interception of the service request; and  
executing the service request.

Reply to Office Action of April 14, 2009

28. (Previously Presented) The method of claim 1 wherein dynamically controlling the software component further comprises:

preventing execution of the service request.

29. (Previously Presented) The method of claim 1 wherein dynamically controlling the software component further comprises:

simulating execution of the service request by returning a value to the software component.

30. (Previously Presented) The method of claim 1 wherein dynamically controlling the software component further comprises:

preventing code that executes in response to interception of the service request from accessing at least some data.

31. (Original) The method of claim 30 further comprising:

allowing code that executes in response to interception of the service request to access alternative data, different from requested data.

32. (Original) The method of claim 31 wherein:

the alternative data comprises a copy of at least some data.

33. (Previously Presented) The method of claim 1 wherein dynamically controlling the software component further comprises:

returning an alternative value to the software component.

34. (Original) The method of claim 33 further comprising:

creating the alternative value by modifying a value returned by the service request.

35. (Previously Presented) The method of claim 1 further wherein the selected desired behavior for the software component further comprises:

preventing code that executes in response to interception of the service request from accessing a system resource.

36. (Original) The method of claim 35 wherein:  
the system resource comprises a network.

37. (Original) The method of claim 35 wherein:  
the system resource comprises storage media.

38. (Original) The method of claim 35 wherein:  
the system resource comprises a file system.

39. (Original) The method of claim 35 wherein:  
the system resource comprises a specific file.

40. (Original) The method of claim 35 wherein:  
the system resource comprises configuration information.

41. (Original) The method of claim 40 wherein:  
the configuration information comprises registry data.

42.-43. (Canceled)

44. (Previously Presented) The method of claim 1 further comprising:  
modifying the at least one dynamically alterable condition dependent rule in response to behavior of the software component.

Reply to Office Action of April 14, 2009

45. (Previously Presented) The method of claim 44 wherein modifying the at least one dynamically alterable condition dependent rule in response to the behavior of the software component further comprises:

responding to an attempt by the software component to access specific data, by consulting a rule that specifies that the software component cannot access other data.

46. (Previously Presented) The method of claim 44 wherein modifying the at least one dynamically alterable condition dependent rule in response to the behavior of the software component further comprises:

responding to an attempt by the software component to access specific data, by consulting a rule that specifies that the software component cannot perform certain functionality.

47. (Previously Presented) The method of claim 1 wherein the at least one dynamically alterable condition dependent rule specifies the desired behavior for the software component and is based on at least one of the following criteria:

- a user with which the software component is associated;
- identity of the software component;
- a time at which the software component is executing;
- history of the software component;
- a source of the software component;
- data which the software component attempts to access;
- functionality that software component attempts to execute; and

computer network resources that the software component attempts to access.

48. (Canceled)

49. (Previously Presented) A computer system for flexibly altering software component behavior, the system comprising:

an interception module, for intercepting a service request made by a software component;

an altered states engine coupled to the interception module, for evaluating the intercepted service request based on at least one dynamically alterable condition dependent rule, an original or modified data in the service request, and at least one of a present software system state and a past software system state, wherein the at least one dynamically alterable condition dependent rule is alterable while the requesting software component is running; and for dynamically selecting at least one desired behavior from among several behaviors for the software component based on the evaluation;

at least one rules database, for storing at least one dynamically alterable condition dependent rule, the rules database being coupled to the altered states engine; and

alternative code for executing in response to an intercepted service request made by the software component, wherein the alternative code is used for dynamically controlling the software component such that the software component executes the selected desired behavior, the alternative code being coupled to the altered states engine.

50. (Previously Presented) A computer system for flexibly altering software component behavior, the system including a processor and memory, comprising:

a software portion for intercepting a service request made by a software component;

a software portion for evaluating the intercepted service request based on at least one dynamically alterable condition dependent rule, an original or modified data in the service request, and at least one of a present software system state and a past software system state, wherein the at least one dynamically alterable condition dependent rule is alterable while the requesting software component is running;

a software portion for dynamically selecting at least one desired behavior from among several behaviors for the software component based on the evaluation; and

a software portion for dynamically controlling the software component such that the software component executes the selected desired behavior, wherein each software portion can be stored in the memory and executed by the processor.

51. (Canceled)



Reply to Office Action of April 14, 2009

52. (Previously Presented) A computer readable medium containing instructions for controlling a processor to perform steps in a method for flexibly altering software component behavior, the steps comprising:

intercepting a service request made by a software component;

evaluating the intercepted service request based on at least one dynamically alterable condition dependent rule, an original or modified data in the service request, and at least one of a present software system state and a past software system, wherein the at least one dynamically alterable condition dependent rule is alterable while the requesting software component is running;

dynamically selecting at least one desired behavior from among several behaviors for the software component based on the evaluation; and

dynamically controlling the software component such that the software component executes the selected desired behavior.

53. (Previously Presented) The computer readable medium of claim 52 wherein intercepting the service request further comprises:

executing alternative code in response to interception of the service request.

54. (Previously Presented) The computer readable medium of claim 52 wherein intercepting the service request further comprises:

preventing code that executes in response to interception of the service request from accessing at least some data.

55. (Previously Presented) The computer readable medium of claim 54 further controlling a processor to perform a step comprising:

allowing code that executes in response to interception of the service request to access alternative data, different from requested data.

56. (Previously Presented) The computer readable medium of claim 52 wherein intercepting the service request further comprises:

passing alternative parameters to code that executes in response to interception of the service request.

57. (Previously Presented) The computer readable medium of claim 56 further controlling a processor to perform a step comprising:

creating the alternative parameters by modifying original parameters passed to the service request.

58. (Previously Presented) The computer readable medium of claim 52 wherein intercepting the service request further comprises:

executing alternative code in response to interception of the service request.

59. (Previously Presented) The computer readable medium of claim 58 further controlling a processor to perform a step comprising:

executing alternative code in response to interception of the service request; and  
executing the service request.

60. (Previously Presented) The computer readable medium of claim 52 wherein intercepting the service request further comprises:

preventing execution of the service request.

61. (Previously Presented) The computer readable medium of claim 52 wherein intercepting the service request further comprises:

simulating execution of the service request by returning a value to the software component.

62. (Previously Presented) The computer readable medium of claim 52 wherein intercepting the service request further comprises:

returning an alternative value to the software component.

63. (Canceled)

64. (Previously Presented) A computer-implemented method in a computer system for flexibly altering software component behavior, the computer-implemented method comprising:

receiving, by an altered states engine, a service request made by a software component;

evaluating the intercepted service request based on at least one dynamically alterable condition dependent rule, an original or modified data in the service request, and at least one of a present software system state and a past software system state, wherein the at least one dynamically alterable condition dependent rule is alterable while the requesting software component is running;

dynamically selecting at least one desired behavior from among several behaviors for the software component based on the evaluation; and

dynamically controlling the software component such that the software component executes the selected desired behavior.

65. (Previously Presented) The method of claim 64 further wherein the selected desired behavior for the software component further comprises:

executing alternative code in response to receiving the service request.

66. (Original) The method of claim 65 further comprising:

executing alternative code in addition to calling the service request.

67. (Original) The method of claim 65 wherein:

the alternative code performs an operation with a same purpose as that of the service request.

68. (Original) The method of claim 65 wherein:

the alternative code performs an operation with a different purpose from that of the service request.

69. (Previously Presented) The method of claim 64 further wherein the selected desired behavior for the software component further comprises:

preventing execution of the service request.

70. (Original) The method of claim 69 further comprising:

returning a value to the software component so as to simulate execution of the service request, without actually calling the service request.

71. (Previously Presented) The method of claim 64 further wherein the selected desired behavior for the software component further comprises:

preventing code that executes in response to receipt of the service request from accessing at least some data.

72. (Original) The method of claim 71 further comprising:

allowing code that executes in response to receipt of the service request to access alternative data, different from requested data.

73. (Previously Presented) The method of claim 72 wherein:

the alternative data comprises a copy of at least some requested data.

74. (Original) The method of claim 71 wherein:

code that executes in response to the receipt of the service request comprises at least alternative code.

75. (Original) The method of claim 71 wherein:

code that executes in response to the receipt of the service request comprises at least the service request.

76. (Previously Presented) The method of claim 64 wherein dynamically controlling the software component further comprises:

passing alternative parameters to code that executes in response to receipt of the service request.

77. (Original) The method of claim 76 further comprising:

creating the alternative parameters by modifying original parameters passed to the service request.

78. (Previously Presented) The method of claim 64 wherein dynamically controlling the software component further comprises:

executing alternative code in response to receipt of the service request.

79. (Previously Presented) The method of claim 64 wherein dynamically controlling the software component further comprises:

executing alternative code in response to receipt of the service request; and  
executing the service request.

80. (Previously Presented) The method of claim 64 wherein dynamically controlling the software component further comprises:

preventing execution of the service request.

81. (Previously Presented) The method of claim 64 wherein dynamically controlling the software component further comprises:

simulating execution of the service request by returning a value to the software component.

82. (Previously Presented) The method of claim 64 wherein dynamically controlling the software component further comprises:

preventing code that executes in response to receipt of the service request from accessing at least some data.

83. (Original) The method of claim 82 further comprising:

allowing code that executes in response to receipt of the service request to access alternative data, different from requested data.

84. (Original) The method of claim 83 wherein:  
the alternative data comprises a copy of at least some data.
85. (Previously Presented) The method of claim 64 wherein dynamically controlling the software component further comprises: returning an alternative value to the software component.
86. (Original) The method of claim 85 further comprising:  
creating the alternative value by modifying a value returned by the service request.
87. (Previously Presented) The method of claim 64 further wherein the selected desired behavior for the software component further comprises:  
preventing code that executes in response to receipt of the service request from accessing a system resource.
88. (Original) The method of claim 87 wherein:  
the system resource comprises a network.
89. (Original) The method of claim 87 wherein:  
the system resource comprises storage media.
90. (Original) The method of claim 87 wherein:  
the system resource comprises a file system.
91. (Original) The method of claim 87 wherein:  
the system resource comprises a specific file.

92. (Original) The method of claim 87 wherein:  
the system resource comprises configuration information.

93. (Original) The method of claim 92 wherein:  
the configuration information comprises registry data.

94.-95. (Canceled)

96. (Previously Presented) The method of claim 64 further comprising:  
modifying the at least one dynamically alterable condition dependent rule in  
response to behavior of the software component.

97. (Previously Presented) The method of claim 96 wherein modifying the at least  
one dynamically alterable condition dependent rule in response to the behavior of the  
software component further comprises:  
responding to an attempt by the software component to access specific data by  
consulting a rule that specifies that the software component cannot access other data.

98. (Previously Presented) The method of claim 96 wherein modifying the at least  
one dynamically alterable condition dependent rule in response to the behavior of the  
software component further comprises:  
responding to an attempt by the software component to access specific data by  
consulting a rule that specifies that the software component cannot perform certain  
functionality.

99. (Previously Presented) The method of claim 64 wherein the at least one  
dynamically alterable condition dependent rule specifies the desired behavior for the  
software component and is based on at least one of the following criteria:  
a user with which the software component is associated;  
identity of the software component;  
a time at which the software component is executing;

Reply to Office Action of April 14, 2009

- history of the software component;
- a source of the software component;
- data which the software component attempts to access;
- functionality that software component attempts to execute; and

computer network resources that the software component attempts to access.

100. (Canceled)

101. (Previously Presented) A computer system for flexibly altering software component behavior, the computer system comprising:

- a receiving module, for receiving a service request made by a software component;

- an altered states engine coupled to the receiving module and controlled by a processor, for:

- evaluating the intercepted service request based on at least one dynamically alterable condition dependent rule, an original or modified data in the service request, and at least one of a present software system state and a past software system state, wherein the at least one dynamically alterable condition dependent rule is alterable while the requesting software component is running; and

- dynamically selecting a desired behavior from among several behaviors for the software component based on the evaluation;

- at least one rules database, for storing at least one dynamically alterable condition dependent rule, the rules database being coupled to the altered states engine; and

- alternative code for executing in response to a received service request made by the software component, wherein the alternative code is used for dynamically controlling the software component such that the software component executes the selected desired behavior, the alternative code being coupled to the altered states engine.



Reply to Office Action of April 14, 2009

102. (Previously Presented) A computer system for flexibly altering software component behavior, the system including a processor and memory, comprising:

a software portion controlled by the processor for receiving a service request made by a software component;

a software portion controlled by the processor for evaluating the intercepted service request based on at least one dynamically alterable condition dependent rule, an original or modified data in the service request, and at least one of a present software system state and a past software system state, wherein the at least one dynamically alterable condition dependent rule is alterable while the requesting software component is running;

a software portion controlled by the processor for dynamically selecting a desired behavior from among several behaviors for the software component based on the evaluation; and

a software portion controlled by the processor for dynamically controlling the software component such that the software component executes the selected desired behavior wherein each software portion can be stored in the memory and executed by the processor.

103. (Canceled)

104. (Previously Presented) A computer readable medium containing instructions for controlling a processor to perform steps in a method for flexibly altering software component behavior, the steps comprising:

receiving a service request made by a software component;

evaluating the intercepted service request based on at least one dynamically alterable condition dependent rule, an original or modified data in the service request, and at least one of a present software system state and a past software system state, wherein the at least one dynamically alterable condition dependent rule is alterable while the requesting software component is running;

dynamically selecting a desired behavior from among several behaviors for the software component based on the evaluation; and

dynamically controlling the software component such that the software component executes the selected desired behavior.

105. (Previously Presented) The computer readable medium containing instructions for controlling a processor of claim 104 wherein receiving a service request made by the software component further comprises:

executing alternative code in response to receiving the service request.

106. (Previously Presented) The computer readable medium containing instructions for controlling a processor of claim 104 wherein receiving a service request made by the software component further comprises:

preventing code that executes in response to receipt of the service request from accessing at least some data.

107. (Previously Presented) The computer readable medium containing instructions for controlling a processor of claim 106 further controlling a processor to perform a step comprising:

allowing code that executes in response to receipt of the service request to access alternative data, different from requested data.

108. (Previously Presented) The computer readable medium containing instructions for controlling a processor of claim 104 wherein receiving a service request made by the software component further comprises:

passing alternative parameters to code that executes in response to receipt of the service request.

109. (Previously Presented) The computer readable medium containing instructions for controlling a processor of claim 108 further controlling a processor to perform a step comprising:

creating the alternative parameters by modifying original parameters passed to the service request.

110. (Previously Presented) The computer readable medium containing instructions for controlling a processor of claim 104 wherein receiving a service request made by the software component further comprises:

executing alternative code in response to receipt of the service request.

111. (Previously Presented) The computer readable medium containing instructions for controlling a processor of claim 110 further controlling a processor to perform a step comprising:

executing alternative code in response to receipt of the service request; and  
executing the service request.

112. (Previously Presented) The computer readable medium containing instructions for controlling a processor of claim 104 wherein receiving a service request made by the software component further comprises:

preventing execution of the service request.

113. (Previously Presented) The computer readable medium containing instructions for controlling a processor of claim 104 wherein receiving a service request made by the software component further comprises:

simulating execution of the service request by returning a value to the software component.

114. (Previously Presented) The computer readable medium containing instructions for controlling a processor of claim 104 wherein receiving a service request made by the software component further comprises:

returning an alternative value to the software component.

115. (Canceled)

116. (New) A method in a computer system for flexibly altering software component behavior, the method comprising:

intercepting a service request made by a software component;

evaluating the intercepted service request based on at least one dynamically alterable condition dependent rule, data in the service request, and a software system state, wherein the at least one dynamically alterable condition dependent rule is alterable while the requesting software component is running;

dynamically selecting at least one desired behavior from among several behaviors for the software component based on the evaluation;

dynamically controlling the software component such that the software component executes the selected desired behavior, wherein the selected desired behavior includes executing alternative code using a copy of requested data in order to virtualize execution of the intercepted service request on the requested data; and

altering operation of the intercepted service request based on a result of the virtualized execution of the intercepted service request.

117. (New) The method of claim 116, further comprising:

executing alternative code to directly modify the at least one dynamically alterable condition dependent rule while the requesting software component is running.

118. (New) The method of claim 117, wherein the at least one dynamically alterable condition dependent rule is modified based on a result of the virtualized execution.

119. (New) The method of claim 116, further comprising:

executing alternative code to directly modify historical execution data while the requesting software component is running.

120. (New) The method of claim 116, wherein the evaluating the intercepted service request includes evaluating the intercepted service request based on a present software system state and a past software system state.